Collective Document Classification with Implicit Inter-document Semantic Relationships

Clinton Burford, Steven Bird and **Timothy Baldwin**
Department of Computing and Information Systems
The University of Melbourne, VIC 3010, Australia
clint@burford.co sbird@unimelb.edu.au tb@ldwin.net

Abstract

This paper addresses the question of how document classifiers can exploit implicit information about document similarity to improve document classifier accuracy. We infer document similarity using simple n-gram overlap, and demonstrate that this improves overall document classification performance over two datasets. As part of this, we find that collective classification based on simple iterative classifiers outperforms the more complex and computationally-intensive dual classifier approach.

1 Introduction

In machine learning, there is a rich tradition of research into the two tasks of: (1) "point-wise" classification, where each instance is represented as an independent instance, and the predictive model attempts to learn a decision boundary to capture instances of a given class; and (2) graphical learning and inference, where instances are connected in a graph, and learning/inference take place relative to the graph structure connecting those instances, based primarily on either conditional dependence (i.e. one event is dependent on the outcome of another) or "homophily" (i.e. the tendency for connected instances to share various properties).[1] Various joint models that combine the two have also been proposed, although in natural language processing at least, these have focused largely on conditional dependence, in the form of models such as hidden Markov models (Rabiner and Juang, 1986) and conditional random fields (Lafferty et al., 2001), where independent properties of words, e.g., are combined with conditional dependencies based on their context of use to jointly predict the senses of all words in a given sentence (Ciaramita and Johnson, 2003; Johannsen et al., 2014).

This paper explores the utility of homophily within joint models for document-level semantic classification, focusing specifically on tasks which are not associated with any explicit graph structure. That is, we examine whether *implicit* semantic document links can improve the results of a point-wise (content-based) classification approach.

Explicit inter-document links have been variously shown to improve document classifier performance, based on information sources including hyperlinks in web documents (Slattery and Craven, 1998; Oh et al., 2000; Yang et al., 2002), direct name-references in congressional debates (Thomas et al., 2006; Burfoot et al., 2011; Stoyanov and Eisner, 2012), citations in scientific papers (Giles et al., 1998; Lu and Getoor, 2003; McDowell et al., 2007), and user mentions or retweets in social media (Jiang et al., 2011; Tan et al., 2011). However, document collections often don't contain explicit inter-document links, limiting the practical usefulness of such methods. In this paper, we seek to expand the reach of research which incorporates linking information, in inducing implicit linking information between documents, and demonstrating that the resultant (noisy) network structure improves document classification accuracy.

The intuition underlying this work is that some types of documents have features which are either absent or ambiguous in training data, but which have

[1] In some tasks, it can also indicate heterophily, i.e. the tendency for connected instances to have contrasting properties, as we shall see for one of our two dataset.

*Proceedings of the Fourth Joint Conference on Lexical and Computational Semantics (*SEM 2015)*, pages 106–116,
Denver, Colorado, June 4–5, 2015.

the special characteristic of indicating relationships between the labels of documents. Most often, an inter-document relationship indicates that two documents have the same label, but depending of the task, it may also indicate that they have different labels. In either case, classifiers gain an advantage if they can consider these features as well as conventional content-based features.

The major contribution of this paper is in showing that document classification accuracy can be improved over a range of datasets using automatically-induced implicit semantic inter-document links, using collective classification. We are the first to achieve this using a general-purpose setup, as applied to a range of datasets. Our results are achieved using n-gram overlap features for both the CONVOTE and BITTERLEMONS corpora, without the use of annotations for explicit semantic inter-document relationships. A second contribution of this work is the finding that simple iterative classifiers outperform more complex dual classifiers when using implicit inter-document links. This finding contradicts earlier work using *explicit* document links, where the dual classifier approach has generally been found to perform best (Thomas et al., 2006; Burfoot et al., 2011). While the work presented here is conceptually quite simple, the findings are significant and potentially open the door to accuracy improvements on a range of document-level semantic tasks.

2 Related Work

Previous work has dealt with the question of collective document classification using implicit inter-document relationships in two basic ways:

1. **proximity**: use a spatial or temporal dimension of the domain to relate documents (Agrawal et al., 2003; Goldberg et al., 2007; McDowell et al., 2009; Somasundaran et al., 2009).

2. **similarity**: relate documents via some notion of their content-based similarity (Blum and Chawla, 2001; Joachims, 2003; Takamura et al., 2007; Sindhwani and Melville, 2008; Jurgens, 2013)

The work using similarity-based links is the closest to ours but is also strongly differentiated because it focuses on transductive semi-supervised classification. That task begins with the premise that only a small amount of labelled training data is available, so content-only classification is likely to be inaccurate. By contrast, the supervised techniques in this paper deal with large amounts of labelled training data and relatively high content-only performance – 76% for CONVOTE and 87% for BITTERLEMONS. It is reasonable to assume that the types of similarity-based relationships derived for transductive semi-supervised classification would be ineffective in a supervised context.

This conclusion is supported by an experiment that shows that the vocabularies of document pairs tend to overlap to similar degrees regardless of document class (Pang and Lee, 2005).

3 Corpora

We experiment with two corpora in this research: CONVOTE and BITTERLEMONS. These two are selected on the grounds that they satisfy two intuitive criteria about types of text collections that may contain features that are not useful for content-only classification, but which may indicate relationships between pairs of documents: (1) the corpora both use an unconstrained prose vocabulary, which increases the likelihood that authors will use distinctive words or sequences of words that are not frequent enough to be useful in training, but which can be used to semantically relate pairs of documents (c.f. newswire articles); and (2) the majority of the text content in both corpora is clearly relevant to the dimension of classification, i.e. there is minimal use of "boilerplate" or "background" material, so the pool from which to select task-relevant content to form inter-document semantic relationships is larger.

3.1 CONVOTE

CONVOTE (Thomas et al., 2006) consists of US congressional speeches relating to a specific bill or resolution, and the ultimate vote of each speaker ("for" or "against"). The document classifier uses the text of each speech to predict the vote of the speaker. Three modifications are made to the corpus: (1) speeches by the same speaker are concatenated, to more naturally represent the requirement that each speaker only has one vote; (2) we drop

	Total
Tokens	1.2M
Speeches	1699
Debates	53
Average speakers/speeches per debate	32
Average tokens per speech	735
Proportion of FOR speeches	49%

Table 1: Corpus statistics for CONVOTE.

	Total
Tokens	0.5M
Articles	594
Topics	149
Average articles per topic	4
Average tokens per article	843
Percentage of ISRAELI speeches	50%

Table 2: Corpus statistics for BITTERLEMONS.

the fixed train, test, development set assignments from the original dataset, and instead evaluate using leave-one-out cross-validation over the 53 debates contained in the dataset, to allow for a more statistically robust evaluation; and (3) we discard the manually annotated inter-document relationships based on references to speaker names, because implicit relationships are the focus of this work.

Table 1 gives statistics for our rendering of CONVOTE. The identical figures for the average number of speeches and speakers per debate reflect the fact that each speaker now contributes only one unified speech.

3.2 BITTERLEMONS

BITTERLEMONS (Lin et al., 2006) is a collection of articles on the Israeli–Arab conflict harvested from the Bitterlemons website.[2] In each weekly issue, the editors contribute an article giving their perspectives on some aspect of the conflict, and two guest authors contribute articles, one from an Israeli perspective and the other from a Palestinian perspective. Sometimes these guest contributions take the form of an interview, in which case we remove the questions (from the editors) and retain only the answers.

The statistics in Table 2 give a picture of the size and structure of BITTERLEMONS.

In accordance with Lin et al. (2006), we experiment with heldout evaluation, with all articles contributed by the editors placed in the training set and those contributed by the guests in the test set. This allows the task to be framed as "perspective" classification, rather than author attribution, i.e. we are focused on the content of the contributions rather than stylistic or biographical features that may identify one editor or the other.

[2] http:/www.bitterlemons.org/

4 Implicit Inter-document Similarity

To implement the hypothesis that documents that use the same rare word or sequence of words are more likely to carry the same label, we calculate a cosine similarity metric between every pairing of documents in a given corpus, using an idf-weighted term vector used to represent document d_i. The idf weighting serves to emphasise terms that are rare within the corpus, and de-emphasise terms that are common. To further enhance this effect, we represent terms by existence-based rather than frequency-based features.

An example of a (tokenised) high-idf sentence pair from CONVOTE is (with the speaker, party affiliation and vote shown in each case, and the high-idf token underlined):

(1) the president s top counselor dan bartlett said this week that there is no magic wand to reduce gas prices . [CROWLEY, JOE (D); AGAINST]

(2) mr. chairman , yesterday the president said , i wish i could simply wave a magic wand and lower gas prices tomorrow. [EMANUEL, RAHM (D); AGAINST]

An example for BITTERLEMONS is:

(3) Even if we /wanted/ to succumb to Israeli pressure, it is impossible to make a Palestinian teach his child that Jaffa or Haifa or Palestine

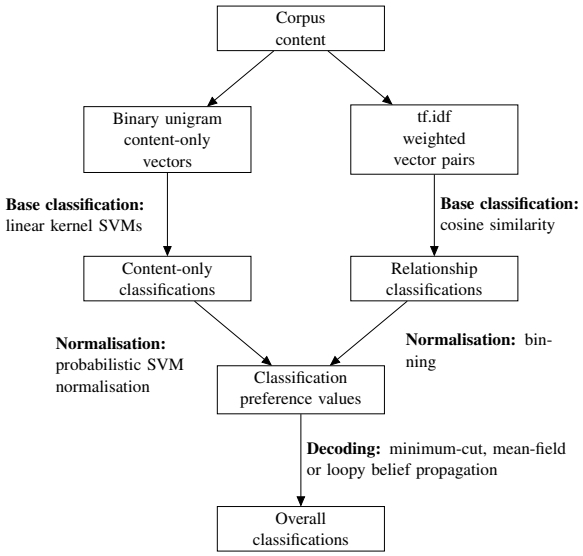


Figure 1: Dual classifier with similarity-based links.

before 1948 was not his land. [AHMAD HARB (GUEST); PALESTINIAN]

(4) This is being neglected and Sharon is having his way in brutalizing the Palestinian people in the hope that they will succumb and abandon their rights. [HAIDAR ABDEL SHAFI (GUEST); PALESTINIAN]

For other examples and more justification of this methodology, see Burford (2013).

5 Collective Classification

Two standard approaches to collective classification are: (1) the dual classifier approach; and (2) the iterative classifier approach. We briefly review these approaches below, but refer the reader to Sen et al. (2008), McDowell et al. (2009) and Burford (2013) for a more detailed methodological discussion.

5.1 Dual Classifier Approach

The dual classifier approach is made up of three steps, as depicted in Figure 1:

1. **Base classification:** Produce base classifications using (1) a **content-only classifier**; and (2) a **relationship classifier**. The content-only classifier makes a binary prediction: FOR and AGAINST for CONVOTE, and ISRAELI or PALESTINIAN for BITTERLEMONS. The relationship classifier indicates the preference that each document pair be SAME or not ($\overline{\text{SAME}}$).

2. **Normalisation:** Normalise the scores, producing values for the classification preference functions, ψ_i, which can be input into a collective classification algorithm.

3. **Decoding:** Produce final classifications by optimally decoding the content-only and relationship level preferences using a collective classification algorithm.

5.1.1 Base classification

For our content-only base classifier, we use the same bag-of-words SVM with binary (existence-based) unigram features as (Thomas et al., 2006). This classifier has been shown to be the best bag-of-words model for BITTERLEMONS (Beigman Klebanov et al., 2010). As our relationship base classifier, we use the cosine similarity scores described above, calculated using n-grams of several different lengths.

5.1.2 Normalisation

We use probabilistic SVM normalisation to convert the signed decision-plane distance output by the content-only classifier into the probability that the instance is in the positive class (Platt, 1999).

For the relationship classifier, the technique used to convert the cosine similarity score into a classification preference needs to fit complex criteria. Preliminary experiments suggested that while the very highest similarity scores are good indicators of SAME relationships, classifier precision drops quickly as recall increases. To avoid polluting the classification graph with large numbers of low-quality links, the normalisation method should incorporate a threshold that discards a significant proportion of the test set pairs. We adopt the following binning technique to convert the cosine similarity score into a probability that the two instances are

SAME:

$$\psi_{ij}(l,l) = \begin{cases} 0.9 & s(i,j) \geq b_1; \\ 0.8 & b_2 \leq s(i,j) < b_1; \\ 0.7 & b_3 \leq s(i,j) < b_2; \\ 0.6 & b_4 \leq s(i,j) < b_3; \\ 0.5 & s(i,j) < b_4; \end{cases}$$

where $\psi_{ij}(l,l)$ represents the SAME preference (i.e. the probability of i and j having the same label); the values for b_1, b_2, b_3, and b_4 are derived by sorting the relationships in the training data by similarity score, and separating them into intervals holding a proportion of SAME pairs equivalent to the nominated probability. This approach is similar to unsupervised discretisation (Kotsiantis and Kanellopoulos, 2006), except the intervals are arranged so that the output categories have a probabilistic interpretation.

5.1.3 Decoding

Decoding is carried out using three techniques: (1) loopy belief propagation (McDowell et al., 2009); (2) mean-field; and (3) minimum-cut.

Loopy Belief Propagation

Loopy belief propagation is a message passing algorithm that can be expressed as:

$$m_{i \to j}(l) = \alpha \sum_{l' \in L} \left(\psi_i(l') \psi_{ij}(l', l) \prod_{k \in N_i \cap D^U \setminus \{j\}} m_{k \to i}(l') \right)$$

$$b_i(l) = \alpha \psi_i(l) \prod_{k \in N_i \cap D^U} m_{k \to i}(l)$$

where $m_{i \to j}$ is a message sent by document d_i to document d_j, and α is a normalization constant that ensures that each message and each set of marginal probabilities sum to 1. The message flow from d_i to d_j communicates the belief of d_i about the label of d_j. The algorithm proceeds by making each node communicate with its neighbours until the messages stabilise. The marginal probability is then derived by calculating $b_i(l)$.

Loopy belief propagation was used in early collective classification work (Taskar et al., 2002) and has remained popular since (Sen et al., 2008; McDowell et al., 2009; Stoyanov and Eisner, 2012).

Mean-field

Mean-field is an alternative message passing algorithm, that can be expressed as:

$$b_i(l) = \alpha \psi_i(l) \prod_{j \in N_i \cap D} \prod_{l' \in L} \psi_{ij}^{b_i(l')}(l', l)$$

and is re-computed for each document until the marginal probabilities stabilise.

Loopy belief propagation and mean-field have both been justified as variational methods for Markov random fields (Jordan et al., 1999; Weiss, 2001; Yedidia et al., 2005).

Minimum Cut

The minimum-cut technique involves formulating a binary collective classification task as a flow graph and finding solutions using standard methods for solving minimum-cut (maximum-flow) problems.

We use the method described by Blum and Chawla (2001) in an in-sample setting, which is equivalent to finding the optimal solution for the cost function for labellings:

$$cost(Y) = \sum_{d_i \in D} w_i(Y_i) + \sum_{(d_i,d_j) \in E : Y_i \neq Y_j} w^r(d_i, d_j)$$

5.1.4 Tuning

The relative weights given to the content-only and relational classifiers can be tuned as follows (for CONVOTE, without loss of generality):

$$\psi'_i(\text{FOR}) = \psi_i(\text{FOR}) + \frac{\min(0,\gamma)(\psi_i(\text{FOR}) - \psi_i(\text{AGAINST}))}{2}$$

$$\psi'_{ij}(\text{FOR}, \text{FOR}) = \psi_{ij}(\text{FOR}, \text{FOR}) - \frac{\max(0,\gamma)(\psi_{ij}(\text{FOR},\text{FOR}) - \psi_{ij}(\text{FOR},\text{AGAINST}))}{2}$$

where ψ'_i and ψ'_{ij} refer to the dampened versions of the content-only and relationship preference functions, respectively, γ is the dampening parameter $\in [-1, 1]$, $\psi'_i(\text{AGAINST}) = 1 - \psi'_i(\text{FOR})$, $\psi'_{ij}(\text{AGAINST}, \text{AGAINST}) = \psi'_{ij}(\text{FOR}, \text{FOR})$, and $\psi'_{ij}(\text{FOR}, \text{AGAINST}) = \psi'_{ij}(\text{AGAINST}, \text{FOR}) = 1 - \psi'_{ij}(\text{FOR}, \text{FOR})$.

This approach works by reducing the difference between the preferences for the two classes (FOR or AGAINST) by an amount that is proportional to the

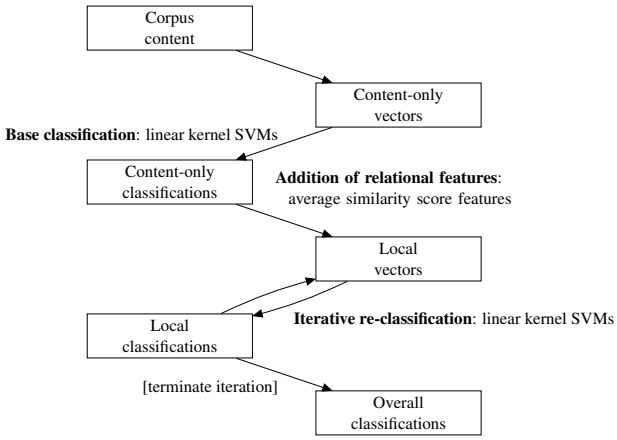


Figure 2: Iterative classifier approach with similarity-based relational features.

absolute value of the dampening parameter. If the dampening parameter is < 0, only the content-only preferences will be dampened (giving more relative weight to relationship preferences). If the dampening parameter is > 0, only the relationship preferences will be dampened (giving more relative weight to the content-only preferences).

For CONVOTE, the training fold is adapted for tuning by use of 52-fold cross-validation, where each of the 52 debates in the training fold is classified using all of the other debates as training data. BITTERLEMONS does not have internal structure within the training set, so it cannot be adapted in this way. Instead, we use leave-one-out cross-validation over the training set. Unfortunately this approach carries the risk of producing base classifications that are unrealistically accurate, because the training set is composed of articles by only two authors.

5.2 Iterative Classifier Approach

The iterative classifier approach has three major components, as depicted in Figure 2:

1. **Base classification**. Produce base classifications using a content-only classifier. As with the dual classifier approach, the content-only classifier will give the preference that each instance be classified with FOR or AGAINST for CONVOTE, and ISRAELI or PALESTINIAN for BITTERLEMONS.

2. **Addition of relational features**. Produce local vectors by adding relational features to the vectors previously used for content-only classification.

3. **Iterative re-classification**. Use a local classifier to classify the new feature vectors. Update the relational features after each iteration to reflect new class assignments. Repeat until class assignments stabilise or a threshold number of iterations is met.

5.2.1 Base Classification

Once again, content-only classification for the iterative classifier is performed using a bag-of-words SVM with binary unigram features.

5.2.2 Relational Features

Let, f^s be an average similarity score:

$$f^s(i,l) = \frac{\sum_{d_j \in D \setminus \{d_i\}} s(i,j)\delta_{Y_j,l}}{\sum_{d_j \in D \setminus \{d_i\}} \delta_{Y_j,l}} \quad (5)$$

where δ is the Kronecker delta. Put in words, f^s is the average of the similarity scores for the pairings of the given instance with each of the instances that have the label l.

We derive relational features for the iterative classifier from the average similarity score as follows:

$$f^{as}(i,l) = \begin{cases} 1 & f^s(i,l) > f^s(i,l'); \\ 0 & \text{otherwise.} \end{cases}$$

This means that the feature $f^{as}(i,l)$ is set to 1 iff the average similarity of document d_i to instances with label l is greater than its average similarity to instances with label l'. In training, document labels are used when counting negative and positive instances to determine the values for f^{as}. In evaluation, the classes assigned in the previous iteration are used.

6 Experiments

We assess the accuracy of the dual classifier and iterative classifier approaches described above over CONVOTE and BITTERLEMONS in terms of classification accuracy, micro-averaging across the 53 folds of cross-validation in the case of CONVOTE. When quoted, statistical significance has been determined

Type	Description	n-gram size				
		1	2	3	4	5
Baseline	Majority	51.44	51.44	51.44	51.44	51.44
Baseline	Content-only	76.40	76.40	76.40	76.40	76.40
Dual	Cosine similarity, min-cut	75.22	77.22⋆	76.52	77.28⋆	77.46⋆
Dual	Cosine similarity, loopy belief	75.10	74.99	75.10	75.46	76.16
Dual	Cosine similarity, mean-field	75.10	74.99	75.10	75.46	76.63
Iterative	Average similarity score	77.99⋆	78.10⋆	78.81⋆	**79.05**⋆	78.16⋆

Table 3: Collective classification performance on CONVOTE (⋆ signifies a statistically significant improvement over the content-only baseline, $p < 0.05$).

Type	Description	n-gram size				
		1	2	3	4	5
Baseline	Majority	49.83	49.83	49.83	49.83	49.83
Baseline	Content-only	86.53	86.53	86.53	86.53	86.53
Dual	Cosine similarity, min-cut	87.88	88.55⋆	88.89⋆	89.90⋆	90.57⋆
Dual	Cosine similarity, loopy belief	87.54	86.87	87.88	87.88	88.55
Dual	Cosine similarity, mean-field	87.54	86.87	87.88	87.88	88.55
Iterative	Average similarity score	87.54	89.90⋆	**90.91**⋆	**90.91**⋆	89.90⋆

Table 4: Collective classification performance on BITTERLEMONS (⋆ signifies a statistically significant improvement over the content-only baseline, $p < 0.05$).

using approximate randomisation with $p < 0.05$ (Nooreen, 1989).

Two baseline scores are shown in the tables for collective classification results: (1) "Majority" gives the performance of the simplest possible classifier, which classifies every instance with the label that is most frequent in training data; and (2) "Content-only" gives the performance of the bag-of-words linear-kernel SVM used to perform base classification.

6.1 Collective Classifier Performance

Table 3 shows the overall collective classifier performance on CONVOTE. The best performer is the iterative classifier with 4-grams, with an accuracy of 79.05%. This is a statistically significant 2.65% absolute gain over the content-only baseline. The iterative classifier is the best performer in general, obtaining the next four best results with statistically significant absolute gains of 2.41%, 1.76%, 1.70% and 1.59% for 3-grams, 5-grams, 2-grams and 1-grams respectively.

The dual classifier with minimum-cut is the next best performer, with a best score of 77.45% for 5-grams, a statistically significant absolute gain of 1.06%. 4-grams and 2-grams also provide statistically significant gains, but 3-grams and 1-grams do not.

For loopy-belief and mean-field the story is less positive. None of the variations gives a statistically significant improvement over the content-only baseline. The best performer is mean-field with 5-grams, with a score of 76.63, a 0.23% absolute improvement over the baseline.

Table 4 shows overall collective classifier performance on BITTERLEMONS. As with CONVOTE, the best performer is the iterative classifier. 4-grams and 3-grams are the top-performing variants, obtaining a score of 90.91%, a statistically significant 4.38% absolute gain over the content-only baseline. 2-grams and 5-grams are the next best, with a statistically significant 3.37% absolute gain over the content-only baseline. 1-grams are the only iterative classifier variant that do not yield a statistically significant improvement over the content-only baseline.

The dual classifier results for BITTERLEMONS

warrant special comment. As mentioned in Section 5.1.4, leave-one-out tuning with the BITTERLEMONS training corpus is compromised. The aim of cross-validation on the training set is to gain a picture of likely performance on the test set. Unfortunately, BITTERLEMONS is not homogeneous: articles in each class in the training set are contributed by just one author, whereas articles in the test set are contributed by different authors. Tuning on BITTERLEMONS failed because leave-one-out on the training set produced 100% accuracy, presumably because there are features specific to the two authors that make classification easy. This meant that the ideal dampening parameter was found to be exactly 1, i.e. collective classification was unnecessary, because the expected performance on the test set was 100%.

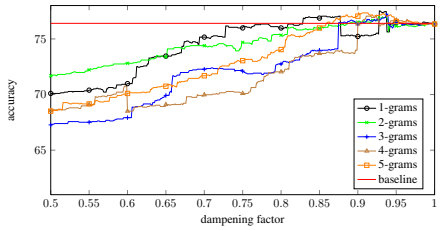
As with CONVOTE, none of the loopy belief or mean-field variants provide statistically significant improvements over the content-only baseline. The best performers are mean-field and loopy belief with 5-grams, with a score of 88.55%, a 2.02% absolute improvement over the baseline.

6.2 Dual Classifier Dampening Response

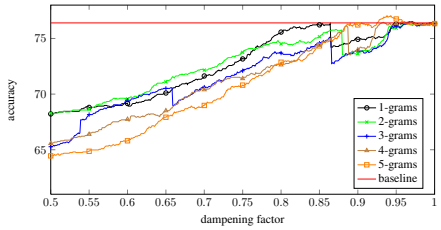
We next examine the dampening response of the dual classifier methods, by presenting six graphs showing the performance of the three different decoding algorithms on the two test corpora. This analysis helps to establish a picture of the limitations of the dual classifier approach in comparison with the iterative classifier approach.

Each of the graphs in this section shows the effect of a varying dampening factor on classification accuracy. In each graph only a small portion of the $[-1, 1]$ range supported by the dampening parameter is shown. The reason for this is visible on many of the graphs: performance is fixed at or near 50% until the dampening parameter is close to 1. This indicates that the probabilities of the content-only classifier and relationship classifier are badly mismatched: performance only becomes reasonable after the relationship preferences have been massively reduced in strength relative to the content-only preferences.

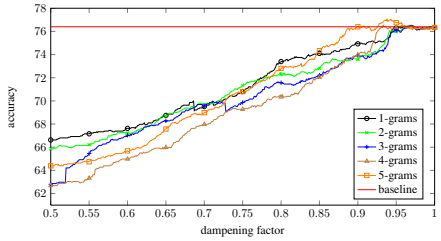
Figure 3 shows performance on CONVOTE for minimum-cut, loopy belief, and mean-field respectively. The trend is the same in each: performance is flat until a sudden jump-up, leading to steady improvement up to a peak, shortly before the maximum dampening value of 1. At 1, the relationship preferences are entirely dampened and performance is the same as the content-only baseline.



(a) Min-cut



(b) Loopy belief



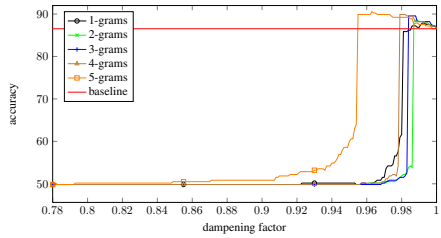
(c) Mean-field

Figure 3: The impact of the dampening factor on dual classifier performance for CONVOTE.

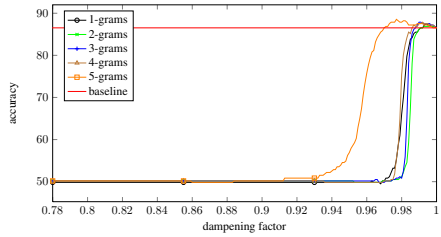
For minimum-cut, 1-grams provide the highest peak accuracy with close to 78% at dampening factor 0.93. Each of the other n-gram orders jumps above the 76.40% baseline at close to this point, with 5-grams providing the most sustained period of high performance from dampening factor 0.85 through to almost 1.

Performance is worse for loopy belief and mean-field. Only 5-grams do better than the baseline, between approximately 0.92 and 0.95 dampening factor for both algorithms.

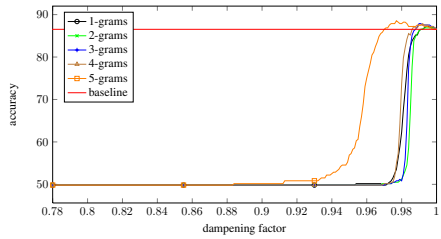
Figure 4 shows performance on BITTERLEMONS



(a) Min-cut



(b) Loopy belief



(c) Mean-field

Figure 4: The impact of the dampening factor on dual classifier performance for BITTERLEMONS.

for minimum-cut, loopy belief, and mean-field respectively. The trend is the same: after a period of flat performance, scores steadily improve as the dampening factor is increased, reaching a peak shortly before the maximum dampening value of 1.

For minimum-cut, 5-grams give the best performance with a peak of 90.57% accuracy at dampening factor 0.95. 4-grams do the next best, followed by 3-grams, 2-grams and 1-grams. Each algorithm rises to a sudden peak and then trails off as it approaches maximum dampening. Loopy belief and mean-field give almost identical performance. Both show the same peak-and-trail-off shape as with minimum-cut but the performance gain is smaller, with 5-grams obtaining a best score of 88.55%.

7 Conclusion and Future Work

The collective classification experiments in this paper demonstrate that useful inter-document semantic relationships can be accurately predicted using features based on *matching sequences of words*, i.e. semantic relationships between pairs of documents that can be detected based on the mutual use of particular n-grams. These semantic relationships can be used to build collective classifiers that outperform standard content-based classifiers.

Iterative classifiers do better than dual classifiers at collective classification using similarity-based relationships. Their superiority goes beyond measures of performance: iterative classifiers are simpler to implement, and more efficient. The key advantage of the iterative classifier seems to lie in its ability to sum up relationship information in a single average similarity score.

Future work should consider the combination of the methods investigated in this paper with more advanced content-only approaches. For dual classifiers and iterative classifiers, it would be also interesting to explore whether alternative base classifiers can provide better performance. For example, confidence-weighted linear classification has been shown to be highly effective on non-collective document classification tasks, and could be easily adapted for use in a dual classifier or iterative classifier (Dredze et al., 2008). Finally, there is significant scope to apply the techniques in this paper to other collective classification tasks and to unambiguously define the types of content for which collective document classification with implicit inter-document relationships can be expected to provide performance gains.

Acknowledgements

This research was supported in part by the Australian Research Council.

References

Rakesh Agrawal, Sridhar Rajagopalan, Ramakrishnan Srikant, and Yirong Xu. 2003. Mining newsgroups using networks arising from social behavior. In *Proceedings of the 12th International Conference on World Wide Web*, pages 529–535, Budapest, Hungary.

Beata Beigman Klebanov, Eyal Beigman, and Daniel Diermeier. 2010. Vocabulary choice as an indicator of perspective. In *Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics: Short papers*, pages 253–257, Uppsala, Sweden.

Avrim Blum and Shuchi Chawla. 2001. Learning from labeled and unlabeled data using graph mincuts. In *Proceedings of the 18th International Conference on Machine Learning*, pages 19–26, Williamstown, USA.

Clinton Burfoot, Steven Bird, and Timothy Baldwin. 2011. Collective classification of congressional floor-debate transcripts. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies*, pages 1506–1515, Portland, USA.

Clinton Burford. 2013. *Collective Document Classification Using Explicit and Implicit Inter-document Relationships*. Ph.D. thesis, The University of Melbourne.

Massimiliano Ciaramita and Mark Johnson. 2003. Supersense tagging of unknown nouns in WordNet. In *Proceedings of the 2003 Conference on Empirical Methods in Natural Language Processing*, pages 168–175, Sapporo, Japan.

Mark Dredze, Koby Crammer, and Fernando Pereira. 2008. Confidence-weighted linear classification. In *Proceedings of the 25th International Conference on Machine Learning*, pages 264–271, Helsinki, Finland.

C. Lee Giles, Kurt D. Bollacker, and Steve Lawrence. 1998. Citeseer: an automatic citation indexing system. In *Proceedings of the 3rd ACM Conference on Digital libraries*, pages 89–98, Pittsburgh, USA.

Andrew B. Goldberg, Xiaojin Zhu, and Stephen Wright. 2007. Dissimilarity in graph-based semi-supervised classification. *Journal of Machine Learning Research*, 2:155–162.

Long Jiang, Mo Yu, Ming Zhou, Xiaohua Liu, and Tiejun Zhao. 2011. Target-dependent Twitter sentiment classification. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies*, pages 151–160, Portland, USA.

Thorsten Joachims. 2003. Transductive learning via spectral graph partitioning. In *Proceedings of the International Conference on Machine Learning*, pages 290–297, Washington, USA.

Anders Johannsen, Dirk Hovy, Héctor Martínez Alonso, Barbara Plank, and Anders Søgaard. 2014. More or less supervised supersense tagging of twitter. In *Proceedings of the Third Joint Conference on Lexical and Computational Semantics (*SEM 2014)*, pages 1–11, Dublin, Ireland.

Michael Jordan, Zoubin Ghahramani, Tommi Jaakkola, Lawrence Saul, and David Heckerman. 1999. An introduction to variational methods for graphical models. *Machine Learning*, 37:183–233.

David Jurgens. 2013. That's what friends are for: Inferring location in online social media platforms based on social relationships. In *Proceedings of the 7th International Conference on Weblogs and Social Media (ICWSM 2013)*, pages 273–282, Dublin, Ireland.

Sotiris Kotsiantis and Dimitris Kanellopoulos. 2006. Discretization techniques: A recent survey. In *GESTS International Transactions on Computer Science and Engineering*, volume 32, pages 47–58.

John D. Lafferty, Andrew McCallum, and Fernando C. N. Pereira. 2001. Conditional random fields: Probabilistic models for segmenting and labeling sequence data. In *Proceedings of the 18th International Conference on Machine Learning*, pages 282–289, Williamstown, USA.

Wei-Hao Lin, Theresa Wilson, Janyce Wiebe, and Alexander Hauptmann. 2006. Which side are you on? Identifying perspectives at the document and sentence levels. In *Proceedings of the 10th Conference on Computational Natural Language Learning*, pages 109–116, New York, USA.

Qing Lu and Lise Getoor. 2003. Link-based classification. In *Proceedings of the 20th International Conference on Machine Learning*, pages 496–503, Washington, USA.

Luke McDowell, Kalyan Moy Gupta, and David W. Aha. 2007. Case-based collective classification. In *Proceedings of the 20th International Florida Artificial Intelligence Research Society Conference*, pages 399–404, Key West, USA.

Luke K McDowell, Kalyan Moy Gupta, and David W Aha. 2009. Cautious collective classification. *Journal of Machine Learning Research*, 10:2777–2836.

Eric W. Nooreen. 1989. *Computer Intensive Methods for Testing Hypothesis*. Wiley and Sons Inc., New York, USA.

Hyo-Jung Oh, Sung Hyon Myaeng, and Mann-Ho Lee. 2000. A practical hypertext categorization method using links and incrementally available class information. In *Proceedings of the 23rd Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 264–271, Athens, Greece.

Bo Pang and Lillian Lee. 2005. Seeing stars: Exploiting class relationships for sentiment categorization with respect to rating scales. In *Proceedings of the 43rd Annual Meeting of the Association for Computational Linguistics*, pages 115–124, Ann Arbor, USA.

John C. Platt. 1999. Probabilistic outputs for support vector machines and comparisons to regularized likelihood methods. In Alexander Smola, Peter Bartlett,

and Bernhard Schölkopf, editors, *Advances in Large Margin Classifiers*, pages 61–74. MIT Press, Cambridge, USA.

Lawrence R. Rabiner and Biing-Hwang Juang. 1986. An introduction to hidden markov models. *ASSP Magazine, IEEE*, 3(1):4–16.

Prithviraj Sen, Galileo Mark Namata, Mustafa Bilgic, Lise Getoor, Brian Gallagher, and Tina Eliassi-Rad. 2008. Collective classification in network data. *AI Magazine*, 29(3):93–106.

Vikas Sindhwani and Prem Melville. 2008. Document-word co-regularization for semi-supervised sentiment analysis. In *Proceedings of the 2008 IEEE International Conference on Data Mining*, pages 1025–1030, Washington, USA.

Seán Slattery and Mark Craven. 1998. Combining statistical and relational methods for learning in hypertext domains. In *Proceedings of Inductive Logic Programming, 8th International Workshop*, pages 38–52, Madison, USA.

Swapna Somasundaran, Galileo Namata, Lise Getoor, and Janyce Wiebe. 2009. Opinion graphs for polarity and discourse classification. In *Proceedings of the 2009 Workshop on Graph-based Methods for Natural Language Processing*, pages 66–74, Singapore.

Veselin Stoyanov and Jason Eisner. 2012. Minimum-risk training of approximate CRF-based NLP systems. In *Proceedings of the 2012 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 120–130, Montréal, Canada.

Hiroya Takamura, Takashi Inui, and Manabu Okumura. 2007. Extracting semantic orientations of phrases from dictionary. In *Human Language Technologies 2007: The Conference of the North American Chapter of the Association for Computational Linguistics*, pages 292–299, Rochester, USA.

Chenhao Tan, Lillian Lee, Jie Tang, Long Jiang, Ming Zhou, and Ping Li. 2011. User-level sentiment analysis incorporating social networks. In *Proceedings of the 17th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 1397–1405, San Diego, USA.

Ben Taskar, Pieter Abbeel, and Daphne Koller. 2002. Discriminative probabilistic models for relational data. In *Proceedings of the 18th Conference on Uncertainty in Artificial Intelligence*, pages 485–492, Alberta, Canada.

Matt Thomas, Bo Pang, and Lillian Lee. 2006. Get out the vote: Determining support or opposition from congressional floor-debate transcripts. In *Proceedings of the 2006 Conference on Empirical Methods in Natural Language Processing*, pages 327–335, Sydney, Australia.

Yair Weiss. 2001. Comparing the mean field method and belief propagation for approximate inference in MRFs. In Manfred Opper and David Saad, editors, *Advanced mean field methods: theory and practice*, pages 229–239. MIT Press, Cambridge, USA.

Yiming Yang, Seán Slattery, and Rayid Ghani. 2002. A study of approaches to hypertext categorization. *Journal of Intelligent Information Systems*, 18(2-3):219–241.

Jonathan Yedidia, William Freeman, and Yair Weiss. 2005. Constructing free-energy approximations and generalized belief propagation algorithms. *IEEE Transactions on Information Theory*, 51:2282–2312.