Optimization of Nearest Neighbors: Run Time and Accuracy

Kyle Wong
3400 N. Charles St.
Baltimore, MD 21218
kwong23@jhu.edu

Tifany Yung
3400 N. Charles St.
Baltimore, MD 21218
tyung1@jhu.edu

Abstract

Ligands can react with a particular receptor in one of three ways: bind and activate the receptor, bind and inhibit the receptor, and not bind at all. Whether a ligand is likely to bind to a receptor can be determined by considering the features such as ligand and receptor shapes, charges, and molecular forces. All of the factors can be accounted via FRED (Fast Rigid Exhaustive Docking) which calculates the Chemgauss4 (cg4) scoring function.

Since the relative positions of both the ligand and receptor affect the likelihood of binding, the FRED scores for 500 receptor frames crossed with each of 710 ligands were taken to create a system for predicting whether a particular ligand will bind with the receptor. These 500 frames were used as the features for feature vectors with the ligands as an instances.

The labeled ligand instances and their 500-FRED-score-long feature vectors were used to train a basic ϵ-sphere, nearest neighbors algorithm with low precision and recall results. The algorithm was then improved by selecting features with information gain and testing with different ϵ values. However, the original dataset proved to be too small and limited to get conclusive results. Synthetic data was then generated for testing and improving the nearest neighbors algorithm. Then, the nearest neighbors algorithm was improved using a Lamba Means EM (Expectation Maximization) algorithm to cluster the data points with highly positive results.

1 Introduction

The ϵ-sphere nearest neighbors algorithm is a close relative of the K-nearest neighbors algorithm; both are clustering algorithms designed with the purpose of categorizing data into a certain number of categories, or labels. As an example, for the ligand-receptor data used, the ligands would be labeled with whether or not they bind with the given receptor.

Clustering algorithms work by using nearby points to determine which cluster or label a given point should be placed in based on the labels of the points around it. The only difference between the above-mentioned ϵ-sphere and K nearest neighbors is how to choose the points around the test point to help determine the test point's label. In the K nearest neighbors algorithm, the number K of points closest to the test point are taken, and the test point is assigned the majority label amongst those points; in the ϵ-sphere algorithm, all points that are within a Euclidean distance of ϵ or less from the test point are used and the majority label is used as the prediction.

The general structure of the ϵ-sphere clustering algorithm is the same as any machine learning algorithm: teaching the algorithm using a training dataset, and testing the accuracy of the algorithm with a development dataset. During training of ϵ-sphere clustering, the training data points are simply read in and stored.

During testing, data points are input individually, and the algorithm attempts to predict their label. It does so by taking all data points within Euclidean distance ϵ of the test point and calculating the ma-

jority label amongst those points. The test point is then assigned to that majority label.

There has been some previous research into speeding up K nearest neighbors by using ϵ-sphere. Indyk and Rajeev (1998) first implemented and tested a kd-tree for the ϵ-sphere algorithm and recommended it only for low-dimensional data. More recently, Muja and Lowe attempted to find fast approximates for the nearest neighbors algorithm.

For receptor ligand docking,

2 Methods

Which ML techniques we picked and why they are suited for our data

3 Results

- output
  - what tests did we run
  - what do these results mean

4 Comparison

The original project proposal suggested three milestones for the optimization of nearest neighbors: minimum, expected, and stretch deliverables. The minimum deliverable was to implement a functional ϵ-sphere nearest neighbor algorithm to predict whether or not a ligand will potentially respond and bind to a given receptor and to implement feature selection via information gain. This implementation was achieved and presented in the class SphereNearestNeighborPredictor.java in the "sphere" package of the "neighbors" project; the results and analysis of the run time and accuracy of this implementation is provided above in the Results section.

The expected deliverable was to implement a faster ϵ-sphere algorithm by using divide-and-conquer in lieu of a linear search for the $epsilon$ closest points to a test point, similar to a kd tree. However, this was not implemented because it was found via additional research (Indyk and Rajeev) that usage of kd trees in higher dimensions are no better than brute-force search since most nodes in the kd tree would need to be evaluated anyways.

5 Junk

Manuscripts must be in two-column format. Exceptions to the two-column format include the title, authors' names and complete addresses, which must be centered at the top of the first page (see the guidelines in Subsection 5.4), and any full-width figures or tables . Type single-spaced. Do not number the pages. Start all pages directly under the top margin. See the guidelines later regarding formatting the first page.

The maximum length of a manuscript is eight (8) pages for the main conference, printed single-sided, plus one (1) page for references (see Section 6 for additional information on the maximum number of pages). Do not number the pages.

5.1 Electronically-available resources

NAACL HLT provides this description in LaTeX2e (`naaclhlt2010.tex`) and PDF format (`naaclhlt2010.pdf`), along with the LaTeX2e style file used to format it (`naaclhlt2010.sty`) and an ACL bibliography style (`naaclhlt2010.bst`). These files are all available at `http://naaclhlt2010.isi.edu`. A Microsoft Word template file (`naaclhlt2010.dot`) is also available at the same URL. We strongly recommend the use of these style files, which have been appropriately tailored for the NAACL HLT 2010 proceedings.

5.2 Format of Electronic Manuscript

For the production of the electronic manuscript you must use Adobe's Portable Document Format (PDF). This format can be generated from postscript files: on Unix systems, you can use `ps2pdf` for this purpose; under Microsoft Windows, you can use Adobe's Distiller, or if you have cygwin installed, you can use `dvipdf` or `ps2pdf`. Note that some word processing programs generate PDF which may not include all the necessary fonts (esp. tree diagrams, symbols). When you print or create the PDF file, there is usually an option in your printer setup to include none, all or just non-standard fonts. Please make sure that you select the option of including ALL the fonts. *Before sending it, test your* PDF *by printing it from a computer different from the one where it was created.* Moreover, some word processor may generate very large postscript/PDF files, where each page is rendered as an image. Such images may reproduce poorly. In this case, try al-

ternative ways to obtain the postscript and/or PDF. One way on some systems is to install a driver for a postscript printer, send your document to the printer specifying "Output to a file", then convert the file to PDF.

For reasons of uniformity, Adobe's **Times Roman** font should be used. In LaTeX2e this is accomplished by putting

```
\usepackage{times}
\usepackage{latexsym}
```

in the preamble.

Additionally, it is of utmost importance to specify the **US-Letter format** (8.5in $\times$ 11in) when formatting the paper. When working with `dvips`, for instance, one should specify `-t letter`.

Print-outs of the PDF file on US-Letter paper should be identical to the hardcopy version. If you cannot meet the above requirements about the production of your electronic submission, please contact the publication chairs above as soon as possible.

5.3 Layout

Format manuscripts two columns to a page, in the manner these instructions are formatted. The exact dimensions for a page on US-letter paper are:

- Left and right margins: 1in
- Top margin:1in
- Bottom margin: 1in
- Column width: 3.15in
- Column height: 9in
- Gap between columns: 0.2in

Papers should not be submitted on any other paper size. Exceptionally, authors for whom it is *impossible* to format on US-Letter paper, may format for *A4* paper. In this case, they should keep the *top* and *left* margins as given above, use the same column width, height and gap, and modify the bottom and right margins as necessary. Note that the text will no longer be centered.

5.4 The First Page

Center the title, author's name(s) and affiliation(s) across both columns. Do not use footnotes for affiliations. Do not include the paper ID number assigned during the submission process. Use the two-column format only when you begin the abstract.

Title: Place the title centered at the top of the first page, in a 15 point bold font. (For a complete guide to font sizes and styles, see Table 1.) Long title should be typed on two lines without a blank line intervening. Approximately, put the title at 1in from the top of the page, followed by a blank line, then the author's names(s), and the affiliation on the following line. Do not use only initials for given names (middle initials are allowed). Do not format surnames in all capitals (e.g., "Leacock," not "LEACOCK"). The affiliation should contain the author's complete address, and if possible an electronic mail address. Leave about 0.75in between the affiliation and the body of the first page.

Abstract: Type the abstract at the beginning of the first column. The width of the abstract text should be smaller than the width of the columns for the text in the body of the paper by about 0.25in on each side. Center the word **Abstract** in a 12 point bold font above the body of the abstract. The abstract should be a concise summary of the general thesis and conclusions of the paper. It should be no longer than 200 words. The abstract text should be in 10 point font.

Text: Begin typing the main body of the text immediately after the abstract, observing the two-column format as shown in the present document. Do not include page numbers.

Indent when starting a new paragraph. For reasons of uniformity, use Adobe's **Times Roman** fonts, with 11 points for text and subsection headings, 12 points for section headings and 15 points for the title. If Times Roman is unavailable, use **Computer Modern Roman** (LaTeX2e's default; see section 5.2 above). Note that the latter is about 10% less dense than Adobe's Times Roman font.

5.5 Sections

Headings: Type and label section and subsection headings in the style shown on the present document. Use numbered sections (Arabic numerals) in

order to facilitate cross references. Number subsections with the section number and the subsection number separated by a dot, in Arabic numerals.

Citations: Citations within the text appear in parentheses as (**?**) or, if the author's name appears in the text itself, as Gusfield (**?**). Append lowercase letters to the year in cases of ambiguities. Treat double authors as in (**?**), but write as in (**?**) when more than two authors are involved. Collapse multiple citations as in (**?**; **?**).

References: Gather the full set of references together under the heading **References**; place the section before any Appendices, unless they contain references. Arrange the references alphabetically by first author, rather than by order of occurrence in the text. Provide as complete a citation as possible, using a consistent format, such as the one for *Computational Linguistics* or the one in the *Publication Manual of the American Psychological Association* (**?**). Use of full names for authors rather than initials is preferred. A list of abbreviations for common computer science journals can be found in the ACM *Computing Reviews* (**?**).

The LaTeX and BibTeX style files provided roughly fit the American Psychological Association format, allowing regular citations, short citations and multiple citations as described above.

Appendices: Appendices, if any, directly follow the text and the references (but see above). Letter them in sequence and provide an informative title: **Appendix A. Title of Appendix**.

Acknowledgment sections should go as a last (unnumbered) section immediately before the references.

5.6 Footnotes

Footnotes: Put footnotes at the bottom of the page. They may be numbered or referred to by asterisks or other symbols.[1] Footnotes should be separated from the text by a line.[2] Footnotes should be in 9 point font.

5.7 Graphics

Illustrations: Place figures, tables, and photographs in the paper near where they are first discussed, rather than at the end, if possible. Wide illustrations may run across both columns and should be placed at the top of a page. Color illustrations are discouraged, unless you have verified that they will be understandable when printed in black ink.

Type of Text	Font Size	Style
paper title	15 pt	bold
author names	12 pt	bold
author affiliation	12 pt	
the word "Abstract"	12 pt	bold
section titles	12 pt	bold
document text	11 pt	
abstract text	10 pt	
captions	10 pt	
bibliography	10 pt	
footnotes	9 pt	

Table 1: Font guide.

Captions: Provide a caption for every illustration; number each one sequentially in the form: "Figure 1. Caption of the Figure." "Table 1. Caption of the Table." Type the captions of the figures and tables below the body, using 10 point text.

6 Length of Submission

The NAACL HLT 2010 main conference accepts submissions of long papers and short papers. The maximum length of a long paper manuscript is eight (8) pages of content and one (1) additional page of references *only* (appendices count against the eight pages, not the additional one page). The maximum length of a short paper manuscript is four (4) pages including references. For both long and short papers, all illustrations, references, and appendices must be accommodated within these page limits, observing the formatting instructions given in the present document. Papers that do not conform to the specified length and formatting requirements are subject to be rejected without review.

References

Cover, Thomas, and Peter Hart. 1967. Nearest neighbor pattern classification. *Information Theory, IEEE Transactions*, 13(1):21-27.

Indyk, Piotr, and Rajeev Motwani. 1998. Approximate nearest neighbors: towards removing the curse

[1] This is how a footnote should appear.

[2] Note the line separating the footnotes from the text.

of dimensionality. *Proceedings of the thirtieth annual ACM symposium on Theory of computing.*

Muja, Marius, and David G. Lowe. 2009. Fast Approximate Nearest Neighbors with Automatic Algorithm Configuration. *VISAPP*, 2009.

Fukunaga, Keinosuke, and Patrenahalli M. Narendra. 1975. A branch and bound algorithm for computing k-nearest neighbors. *Computers, IEEE Transactions*, 100(7): 750-753.